

Comparing Staged and Iterative Processes

20 minutes

Software Engineering and Testing

Outcome 1.19

Outcome 1.23

Outcome 1.5

Outcome 1.8

WHAT THIS LESSON DOES

In this lesson, you'll explore the differences between staged and iterative design processes in software development. You'll recap the basics of each, compare them across key aspects with examples, and summarise the findings in a clear table for reference.

CURRICULUM OUTCOMES

Outcome	What the specification asks for
1.19	identify features of both staged and iterative design and development processes
1.23	reflect and communicate on the design and development process
1.5	evaluate alternative solutions to computational problems
1.8	evaluate the costs and benefits of the use of computing technology in automating processes

WHAT STUDENTS SHOULD BE ABLE TO DO

- Understand the fundamental differences between staged and iterative design processes in software development.
- Analyse key aspects such as structure, flexibility, risk management, delivery, and suitability of each process.
- Evaluate the strengths and limitations of staged and iterative approaches using real-world examples.
- Determine the most appropriate process for specific project types based on their requirements and complexity.
- Develop a clear perspective on how each process impacts project planning, adaptability, and outcomes.

BEFORE THE LESSON

- This is a theory and discussion lesson with no coding or hardware. Nothing needs installing.
- Have a real project in mind for each model to make it concrete: a banking or inventory system for staged, an evolving app or startup product for iterative.

Comparing Staged and Iterative Processes

HOW THE LESSON RUNS

Introduction
Overview of Staged and Iterative Processes
Comparing Structure
Comparing Flexibility
Comparing Risk Management
Comparing Delivery
Comparing Suitable For
Overall Comparison
Wrap Up

TEACHING MOVES

- 2. Overview of Staged and Iterative Processes. Check the recap lands before comparing. Ask students to describe each model in one sentence. If they cannot name the sequential phases of waterfall, the comparison ahead will not stick.
- 3. Comparing Structure. Draw both shapes on the board: a straight line of phases for staged, a repeating loop for iterative. Emphasise that iterative produces a working prototype each cycle, not just documentation.
- 4. Comparing Flexibility. Pose the client-changes-a-feature scenario aloud and let students predict the cost under each model before revealing it. This makes rigidity versus adaptability felt, not just stated.
- 5. Comparing Risk Management. Stress the timing difference: staged concentrates risk at the end, iterative spreads it. Ask why catching a design flaw early is cheaper than after full implementation.
- 6. Comparing Delivery. Link delivery back to risk and flexibility rather than treating it as a separate fact. Early incremental delivery is what generates the feedback that reduces risk.
- 7. Comparing Suitable For. Give students two or three sample projects and have them argue which model fits. The reasoning matters more than the answer, and it rehearses the exam skill.
- 8. Overall Comparison. Use the table as consolidation, not new material. Have students reconstruct a row from memory before showing it, so they own the comparison rather than copying it.

WHAT TO WATCH FOR

- Students treat iterative as simply better than staged, since it sounds more modern and flexible. Return to the suitability point: regulated, stable-requirement projects genuinely suit staged. Ask them to name a project where the full scope is known upfront and change is rare.
- Students confuse iterative delivery with delivering an unfinished or lower-quality product. Clarify that each iteration produces a working, tested version of a portion, not a broken whole. Quality is maintained per cycle; scope grows over time.

DIFFERENTIATION

- Emerging: Give one anchor example per model and ask the student to keep matching new aspects to those two familiar projects rather than juggling abstractions.
- Developing: Have them fill in a blank comparison table from the discussion before the summary is shown, then check against it.